

FIG.1A

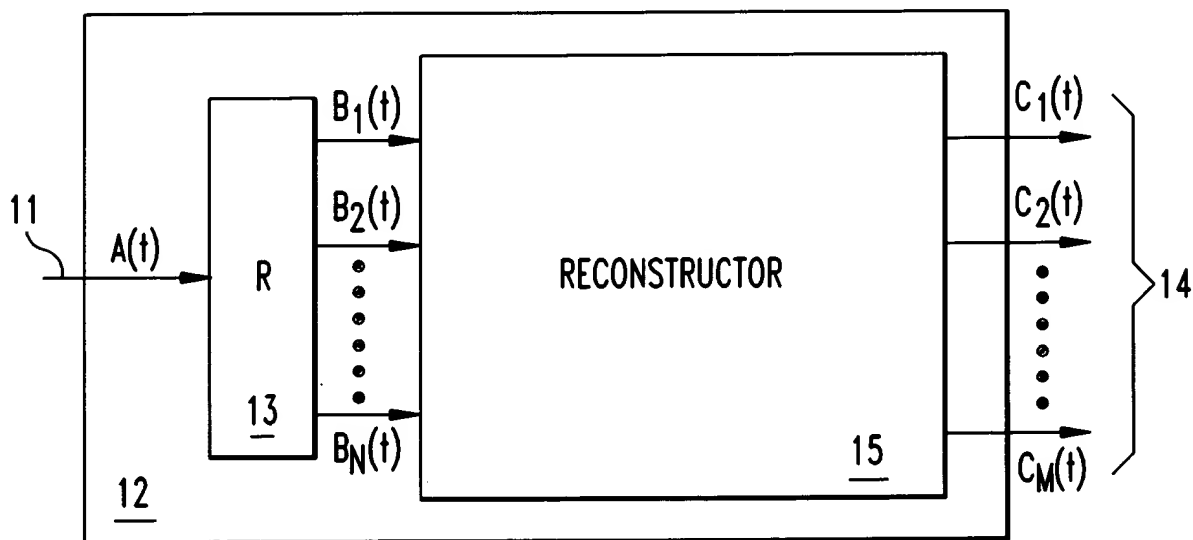


FIG.1B

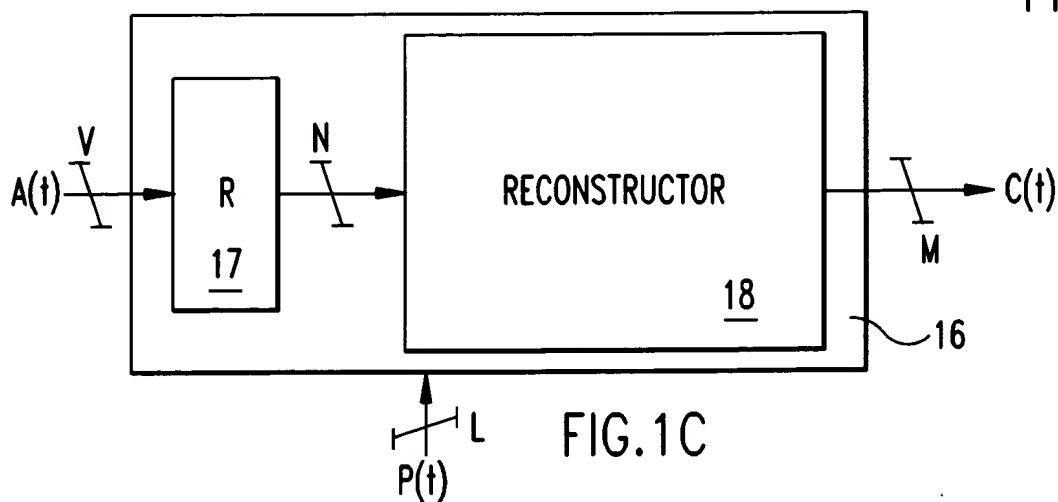


FIG.1C

FIG.2

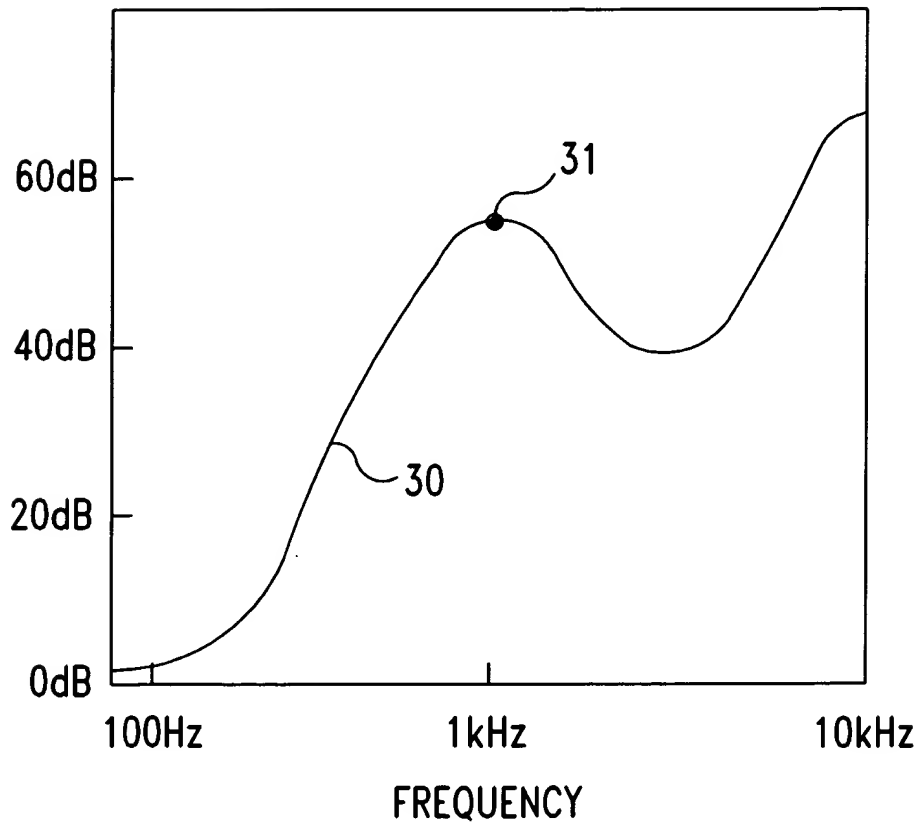
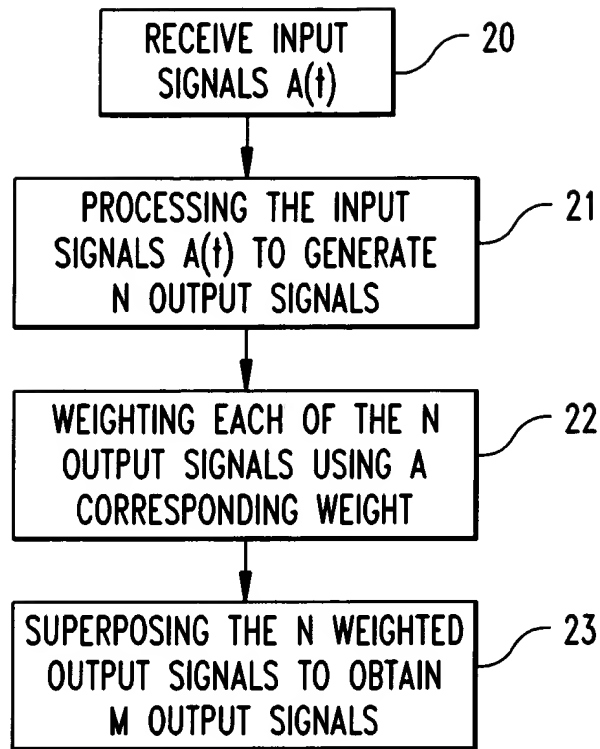


FIG.3

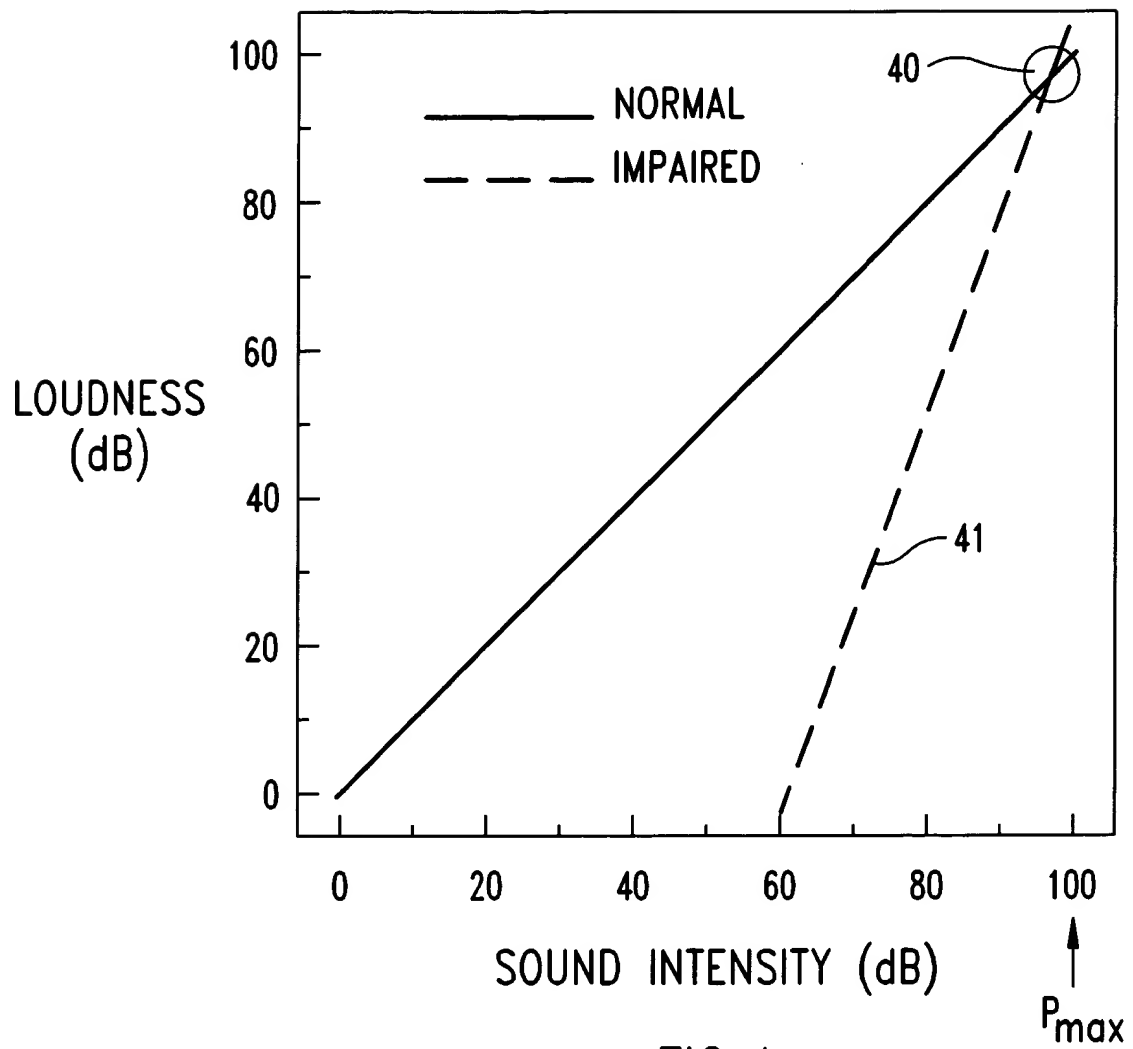


FIG.4

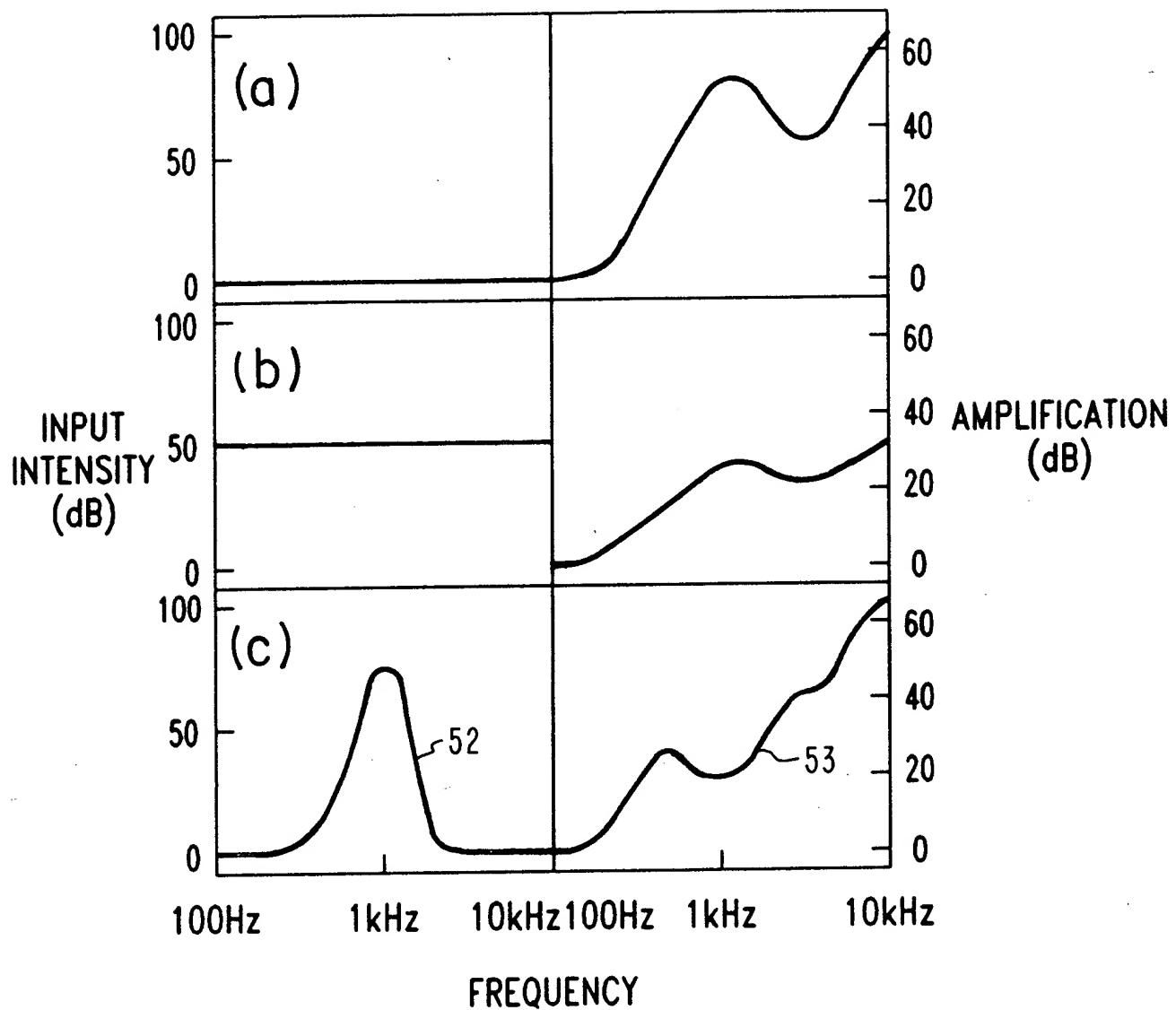


FIG.5

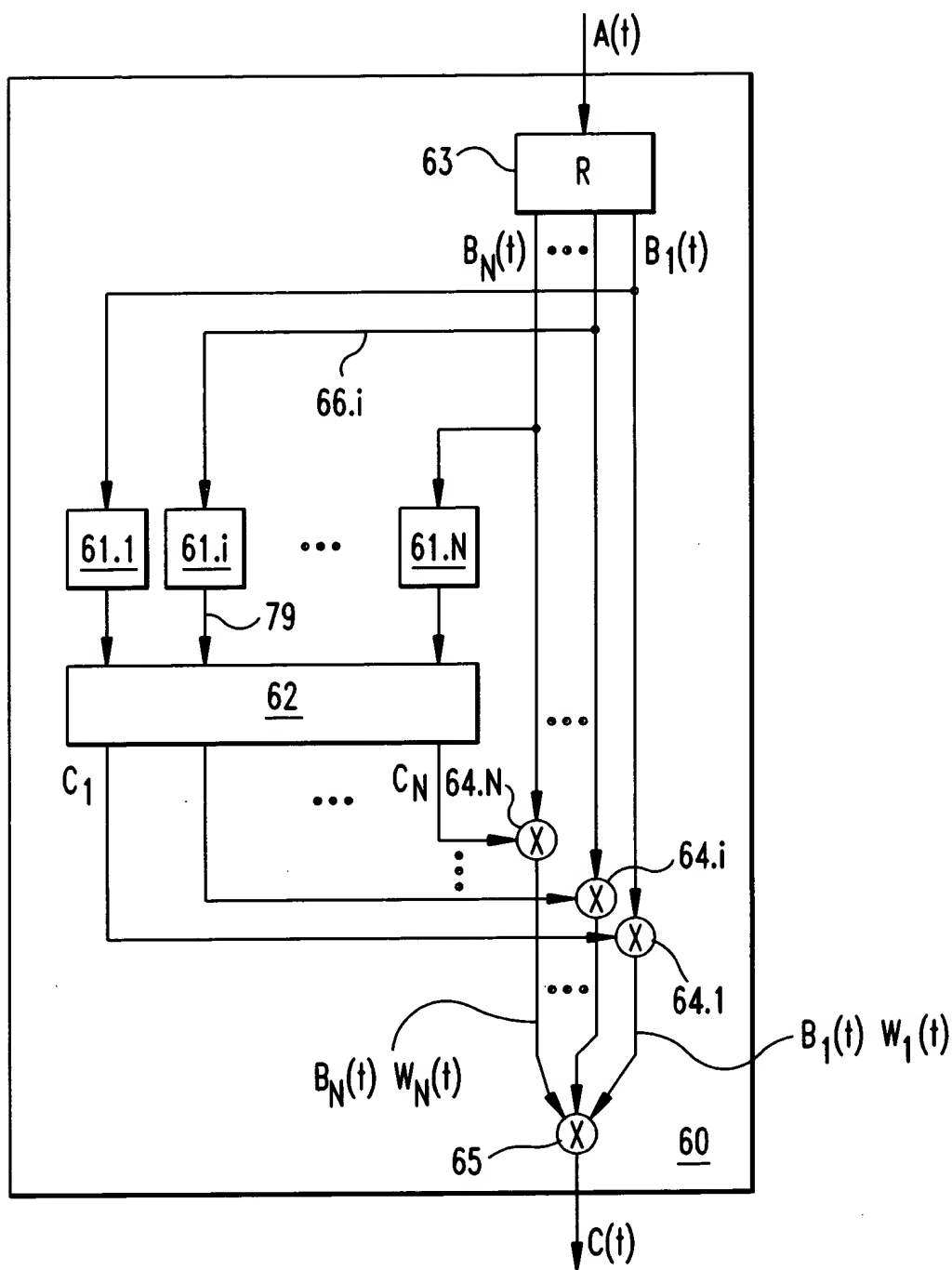


FIG. 6

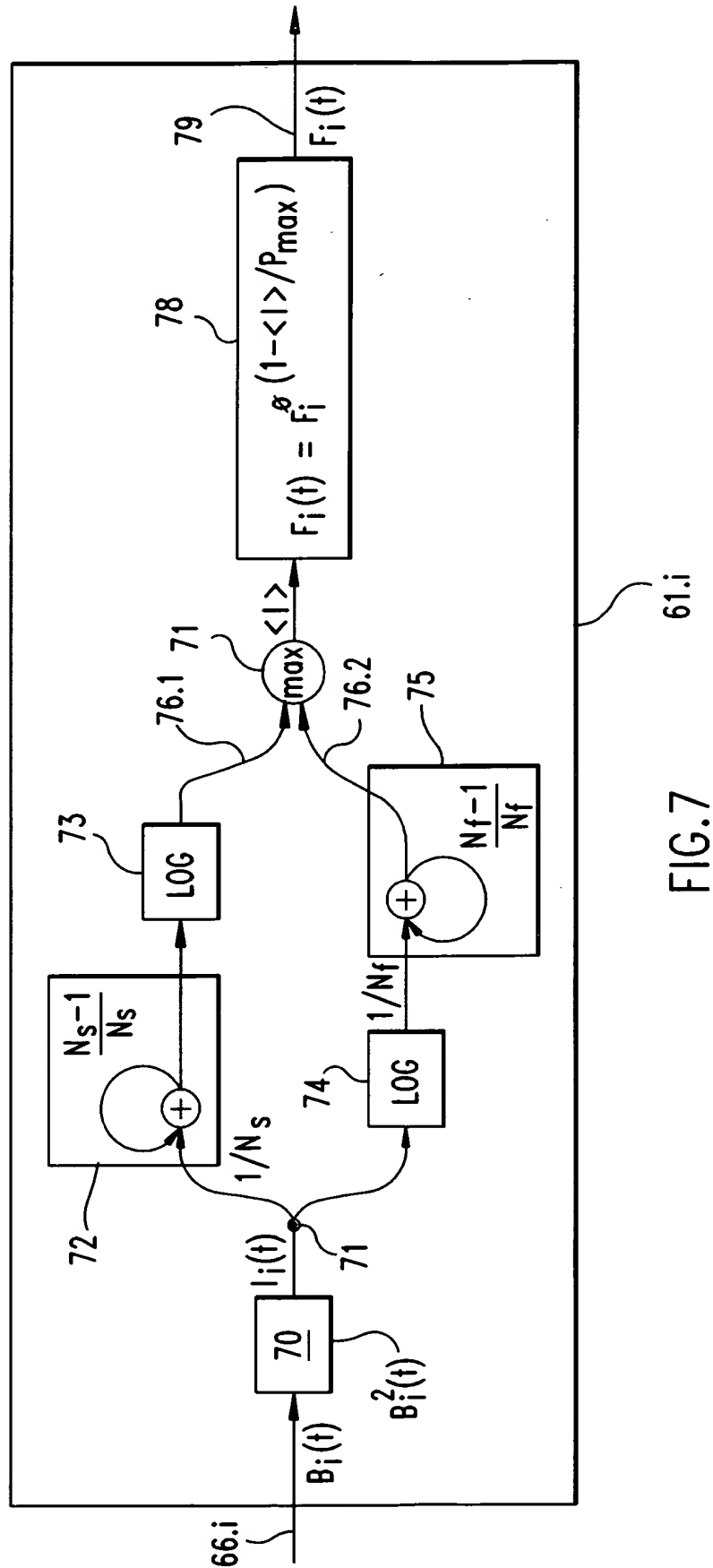
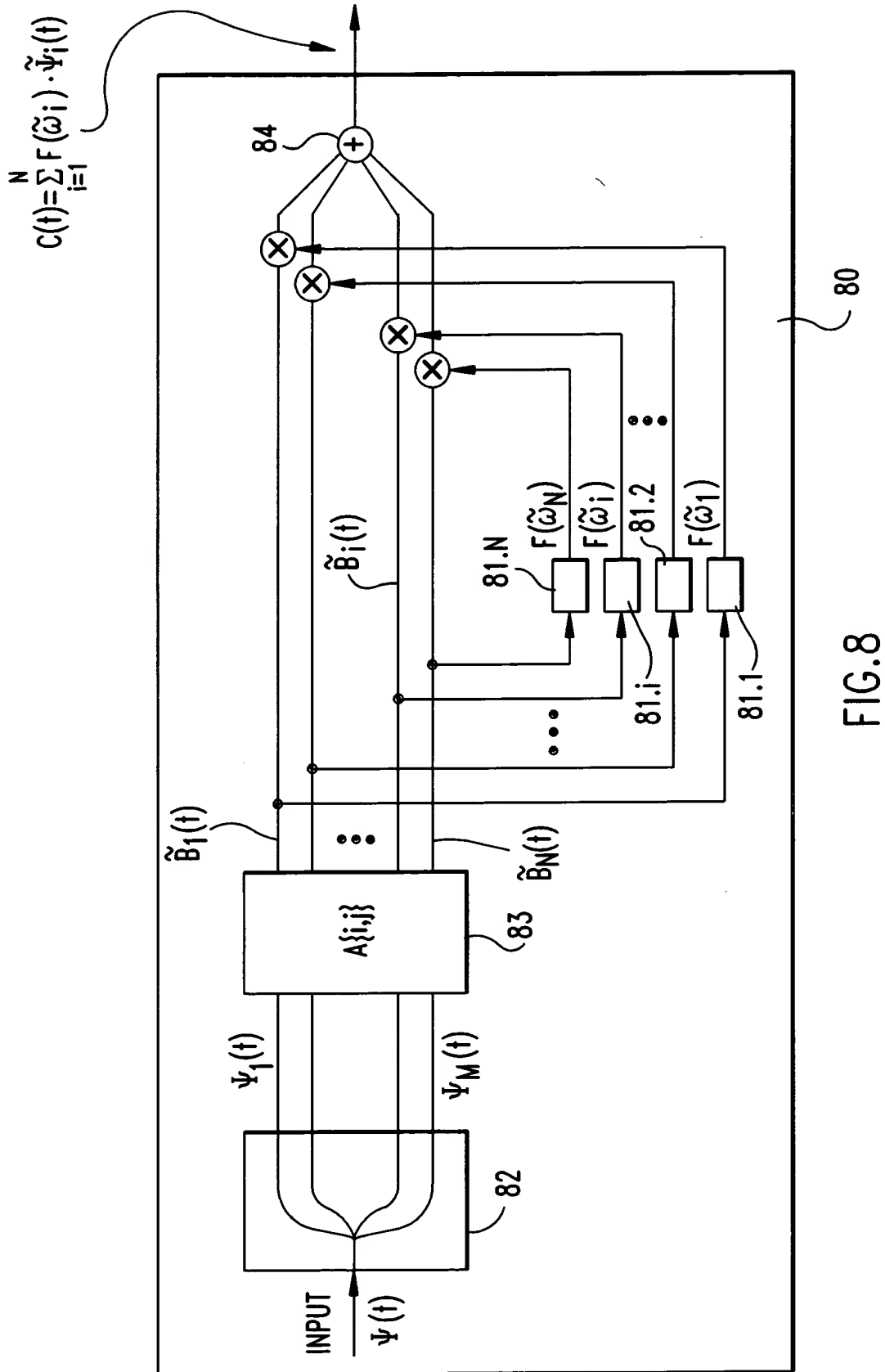


FIG.7



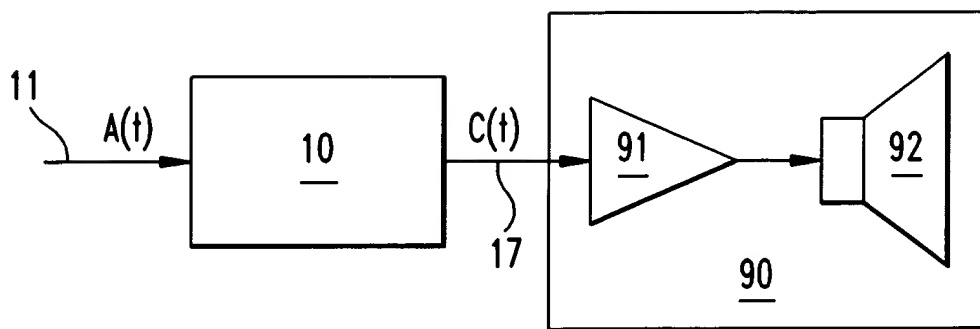


FIG.9

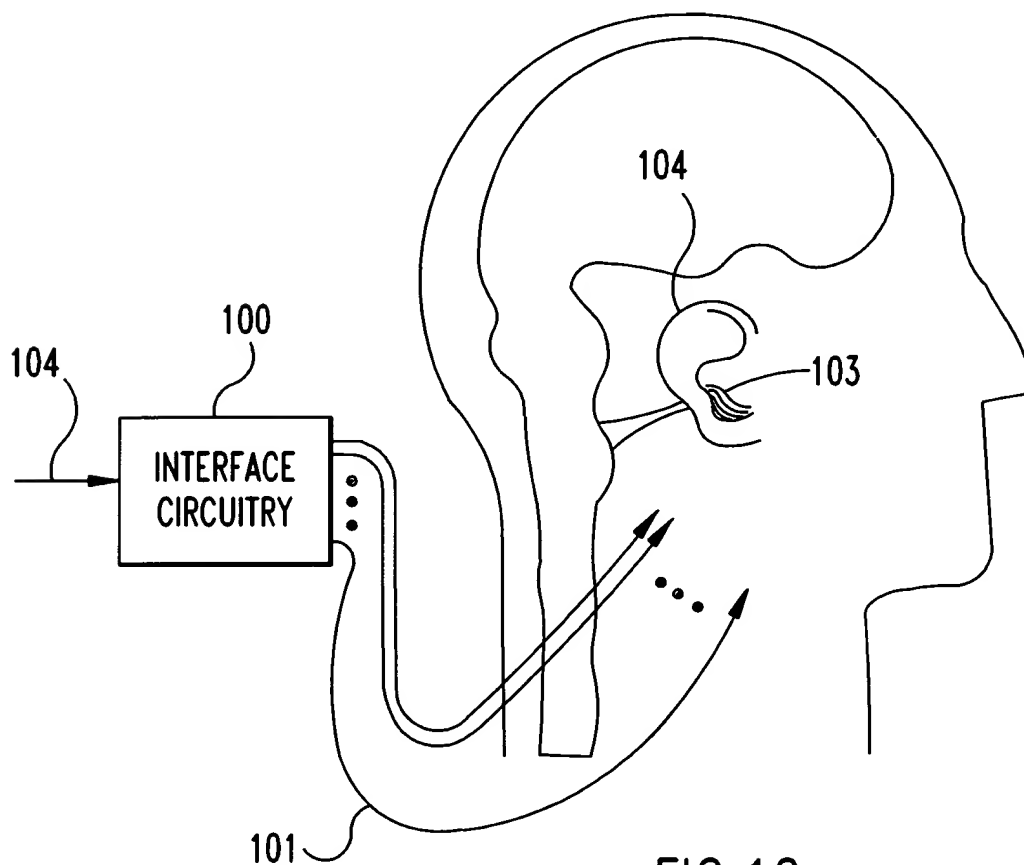


FIG.10

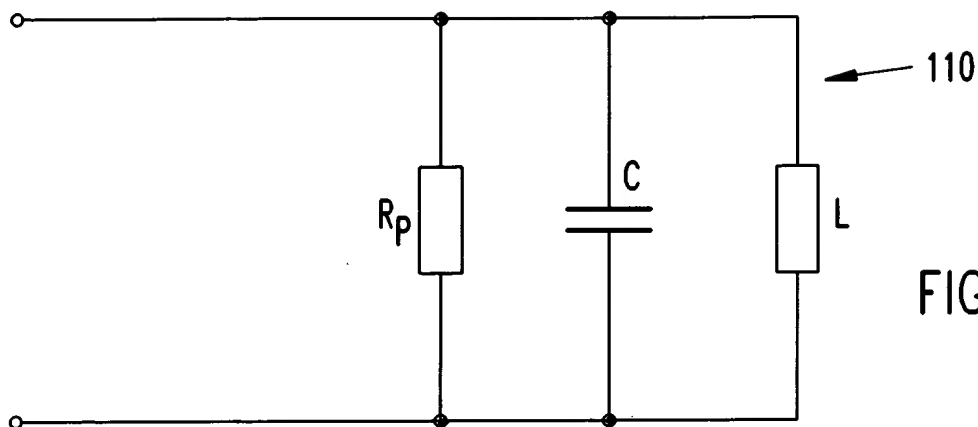


FIG.11

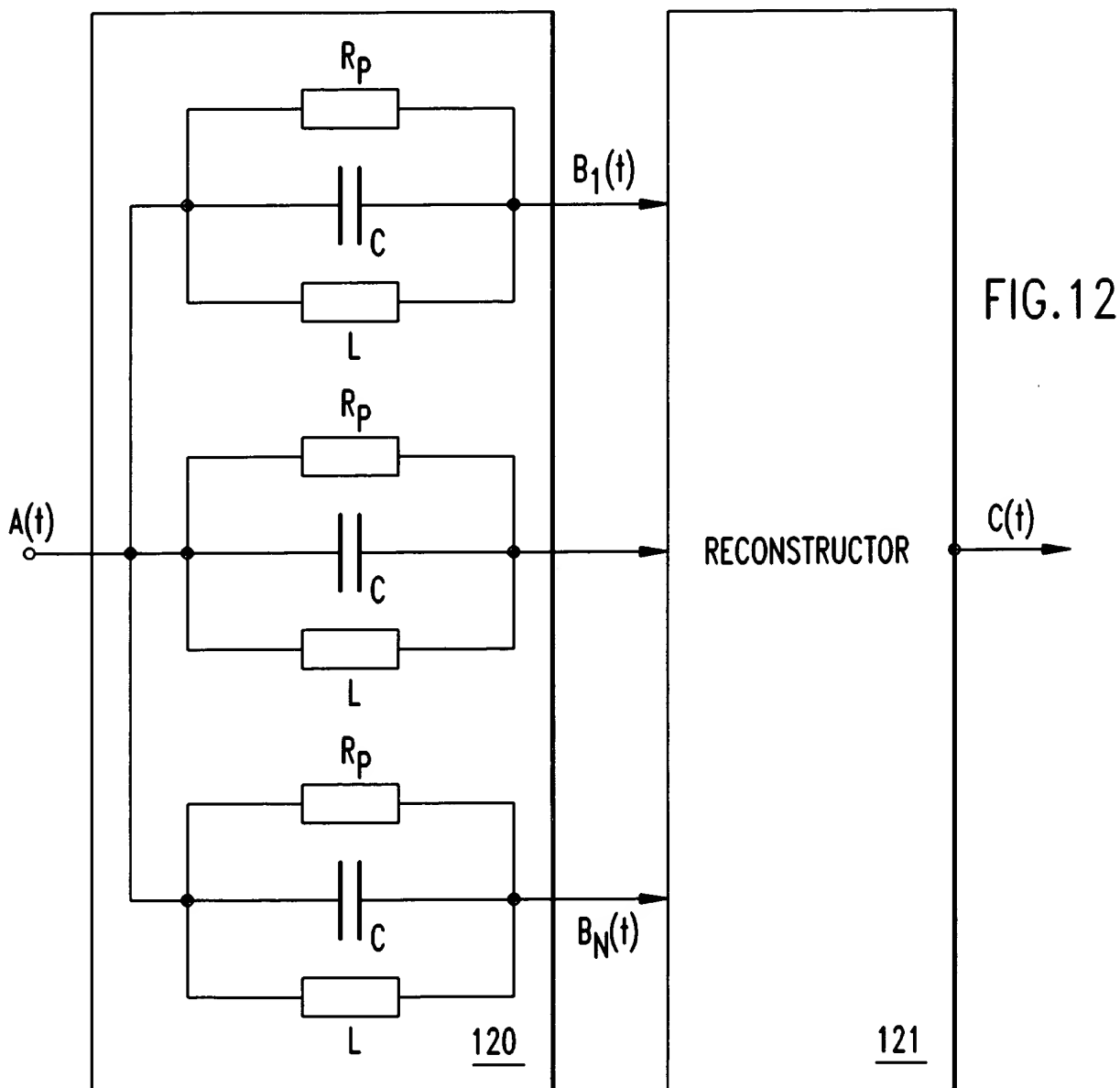


FIG.12



Implementation example:

=====

(read discretization time step)

FIG.13A

get dt

[reset oscillators]

130 for each N [loop over resonators]

get omega(N) [read frequency *2*pi]

get tau(N) [read time decay constant]

get weight(N) [read weight]

$u_o(N) = (2 - (\omega(N) dt)^2) / (1 + dt/\tau(N))$

[one example of equations describing

[a damped harmonic resonator]

$u_m(N) = -(1 - dt/\tau(N)) / (1 + dt/\tau(N))$

$u_i(N) = dt / (1 + dt/\tau(N)) / \tau(N)$

[u are constants]

$b_m(N) = 0$

[b are amplitudes of resonators]

$b_0(N) = 0$

[bm is amplitude of previous time step]

[bo is amplitude of actual time step]

131 done with loop over N

[loop over time steps]

132 for each t

input a_p

[read input signal a of the]

[next time step]

if (first t) then

[the first time, reset stack]

[for input; otherwise a would]

[not be defined]

135 $a_o = a_p$

$a_m = a_p$

end if

[calculate difference aprime of input signal a]

$a_{prime} = (a_p - a_m)$

continue with FIG.13B



FIG.13B

continuation of FIG.13A

[propagate resonators; loop over all resonators]

for each N

134

$$b_p = u_o(N) * b_o(N) + u_m(N) * b_m(N) + u_i(N) * a_{prime}$$

[resonator for following time]

[step]

$$b_m(N) = b_o(N)$$

[stack is updated/shifted]

$$b_o(N) = b_p$$

done with loop over N

[at this point the weights weight (N) can be recalculated]

[as a function of the $b_o(N)$]

[In this implementation example the weights are constant]

[weight and combine resonator output signals to generate]

[output signal c]

$$c = 0$$

for each N

$$c = c + \text{weight}(N) * b_o(N)$$

done with loop over N

[reset stack for input]

$$a_m = a_o$$

$$a_o = a_p$$

[output output signal c]

output c

133

done with loop over t

end.

=====